

زبان R را به E_1 و E_2 قسمت می کنیم.

$$E_1 = \{ a^n b^m \mid n, m \geq 0 \text{ and } n \geq m \}$$

$$E_2 = \{ a^n b^m \mid n, m \geq 0 \text{ and } n < m \}$$

واضح است که اجتماع این زبانها $E = E_1 \cup E_2$

$$\Rightarrow E_1 \cup E_2 = E$$

$$E_1 \cap E_2 = \emptyset$$

$$G_1 \rightarrow aA$$

$$G_2 \rightarrow S_1 \mid S_2$$

$$S_1 \rightarrow aXb$$

$$S_2 \rightarrow aYb$$

$$A \rightarrow aaAb \mid aA \mid \epsilon$$

$$X \rightarrow aXb \mid aaXb \mid aab$$

$$Y \rightarrow aYb \mid Yb \mid b$$

$$G \rightarrow G_1 \mid G_2$$

استنتاج $a^6 b^6$

$$(1) G \rightarrow G_2$$

$$G_2 \rightarrow S_2$$

$$S_2 \rightarrow aYb$$

$$aYb \rightarrow aaYbb$$

$$\Rightarrow - \rightarrow a^6 Y b^6$$

$$a^6 Y b^6 \rightarrow a^6 b b^6 = a^6 b^{12}$$

با هر بار برگرداندن کار یک a به a و یک b به b را اضافه می کنیم

$$M \rightarrow AOMIA$$

$$A \rightarrow 0 \text{ } \cancel{A} \text{ } 1 \text{ } A \text{ } 1 \text{ } \epsilon$$

A- همدی اینتریکت های که تولید می شود و اها در *proliferation* مادی است بدون انجام نیست می کند.

۱- استغفار

$$W, W_1 - W_n$$

$w, w_1 - w_n$
 $c, 101 - 111$

$$n \rightarrow n+1$$

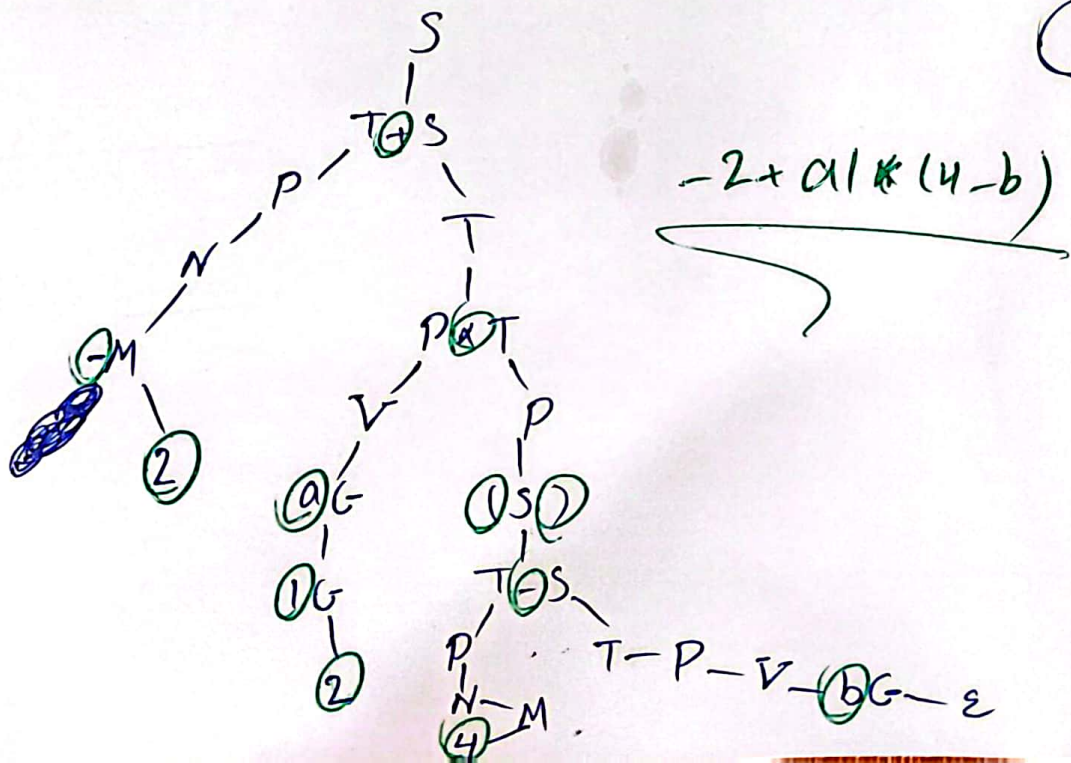
۱. استقامت لا $OA \rightarrow A$ مدخلی حالت این صورتیکه

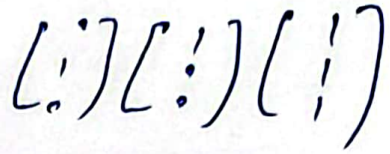
ہمارا حقیقت کرے۔ سب سے دوسرا مرحلہ میں سب پریشان کر رہا ہے۔

استعمالِ سدولِ ابیام موجبِ طبعِ اسکنِ صغیرِ کرکسِ سدولِ ابیام است.

۴۔ حالہ میں تو کہیں کہ ۴۱ ہماری انتہائی حد سے زیادہ ہے۔ دوبارہ مثل قبلی استغفار کرنے سے بڑھ کر جی

فصلی $A \rightarrow 0.A1A$ لذا $M \rightarrow A.0.M$ استعدا کنیم. این فاکتورهای بدون
تظهِیر را تولید می کنند. ما به هم این تولید یک سوشن در پی دستور زبان داده بدون (ابام) است.



$$[\cdot], [\cdot], [\cdot]$$


(;)

مرکز تحقیقات

$$M \rightarrow MM \rightarrow \text{scribble} MMM \rightarrow xyz \quad \checkmark$$

(ب) مہل

$$M \rightarrow \text{var} \mid (S)$$

list & Add (list L , int size , int index)

(الف) (6)

if (index < size - 1)

return ~~list~~ [] ; // empty list

return L [index] + Add (L , size , index + 1)

list swap (list L)

list temp = L [L.Length - 1] + Add (L , L.Length - 1 , 1)

return temp + L [0]

list swap (list L)

int temp = L [~~0~~ L.Length - 1]

L [L.Length - 1] = L [0]

L [0] = temp ,

return L

(ب)

(2) در حالت imperative حافظه اضافی که حالتنا هم داریم - اندک به int که صاف . اما در روش declarative

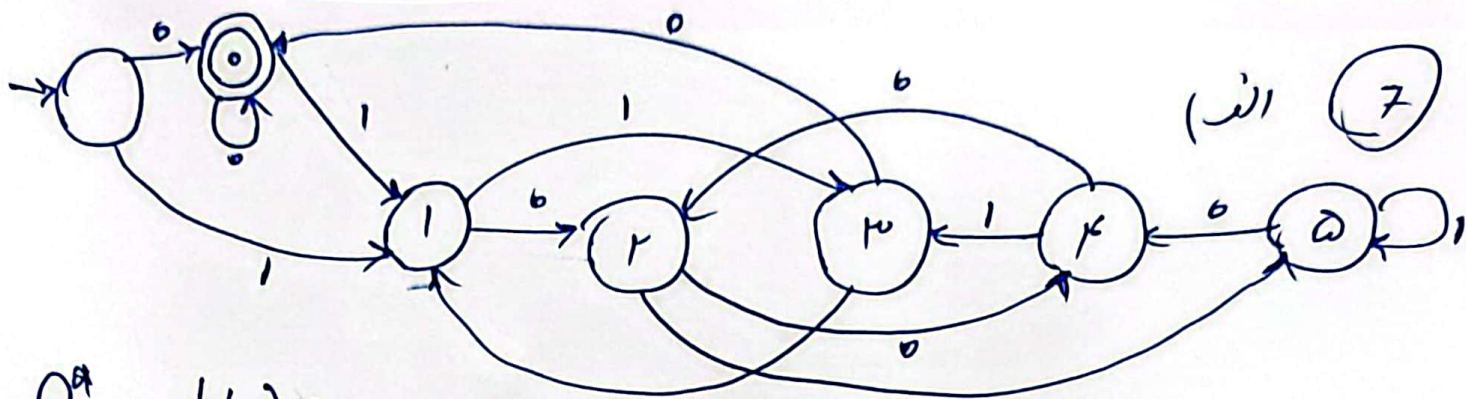
ما هم به دور کل تابع Add به همراه ورودی ها و باید توان اشتک بگیریم . به جای این

کار دو - اندک های طول L (توهمین ارباب) انجام بدیم (رقت حریف یک تیر از طول) که

دو بار هر ورودی فضای خود را می خورد . به دونه هم temp داریم که خودش از ورودی طول بسته

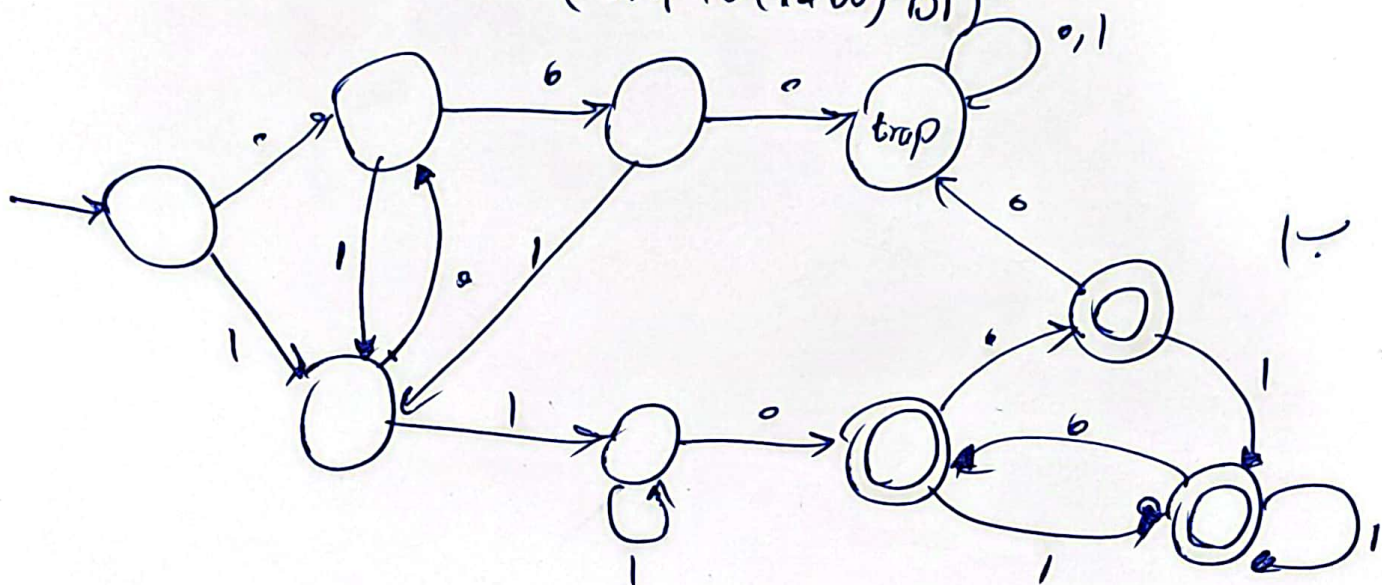
در نهایت می تونه در حدود $O(n^2)$ (اگر n طول لیست باشد) ولی چون طول لیست فقط n

روند هم هست $O(n)$ هست .
 $n + (n-1) + (n+1) = n^2 + n - 1$



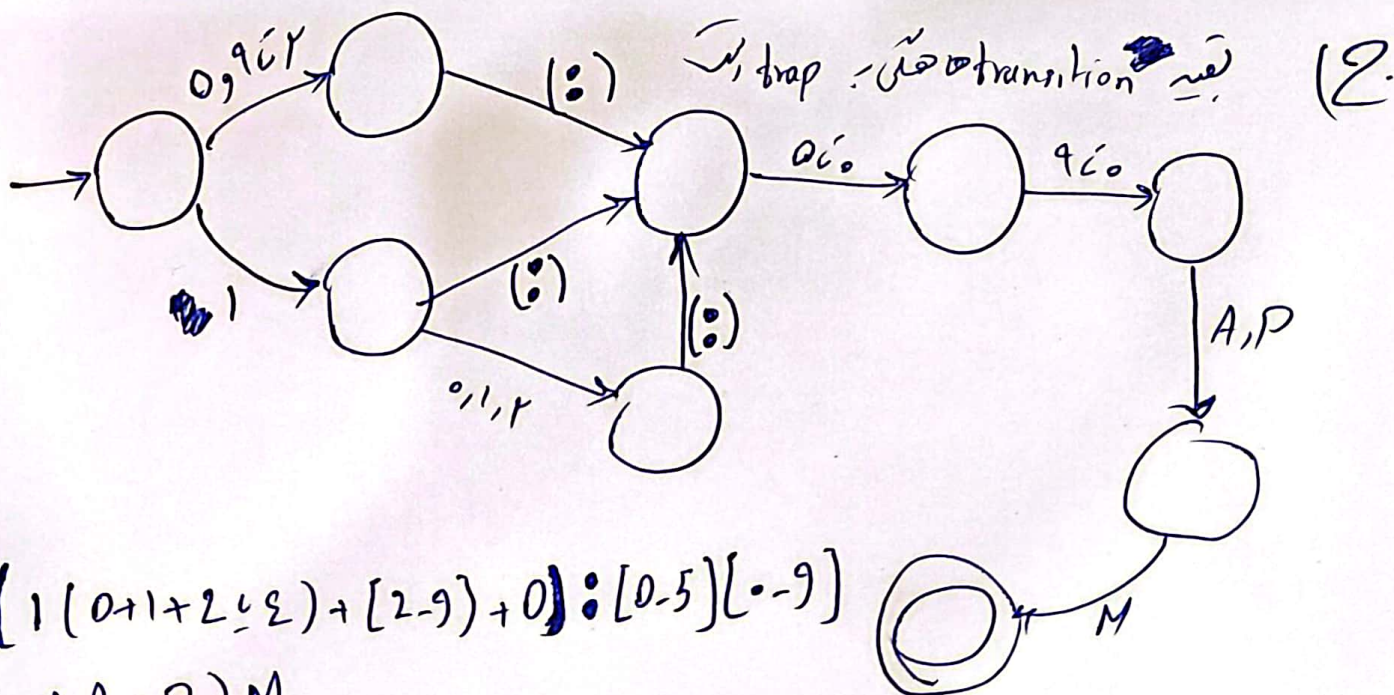
$$0^* + (L_3)0$$

$$L_3 = (11 + 10(1+00)^*01)^*$$



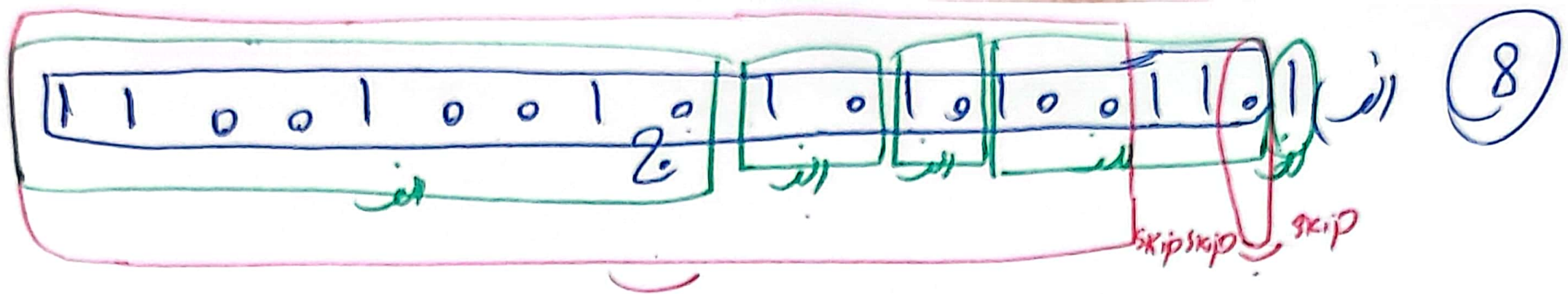
$$(1(01+001)^* + 0(10+010)^*1)(1)^*0(01(1)^*0 + 1(1)^*0)^*$$

$$(0 + 01\varepsilon)$$



$$(1(0+1+2\varepsilon) + [2-9] + 0) : [0-5][\cdot-9]$$

$$(A+P)M$$



حالت جاری زایل ها

